High-Speed, Wide Area, Data Intensive Computing: A Ten Year Retrospective[1]

William E. Johnston
Information and Computing Sciences Division, Lawrence Berkeley National Laboratory
Berkeley, CA, 94720

Abstract

Modern scientific computing involves organizing, moving, visualizing, and analyzing massive amounts of data from around the world, as well as employing large-scale computation. The distributed systems that solve large-scale problems will always involve aggregating and scheduling many resources. Data must be located and staged, cache and network capacity must be available at the same time as computing capacity, etc. Every aspect of such a system is dynamic: locating and scheduling resources, adapting running application systems to availability and congestion in the middleware and infrastructure, responding to human interaction, etc. The technologies, the middleware services, and the architectures that are used to build useful high-speed, wide area distributed systems, constitute the field of data intensive computing. This paper explores some of the history and future directions of that field.

1. Introduction

This paper is a personal view of the evolution of data intensive computing over the past ten years. The evolution is traced through a series of milestones that are based on advances in the technology, architectures, and software, and that have brought us from the point when we were lucky to get a few hundred kilobits/second of application-to-application data on a local area network, to the current time, where we can routinely get almost 500 megabits/second on wide area networks.

This is not a comprehensive review of the field, though I have been involved in many of the seminal activities. I will acknowledge a number of people in the course of this article, but there will be those whose important contributions did not directly intersect our work, and whom I will therefore not mention. The body of this paper is organized into three major sections: where are we today, how did we get there, and where are we going in the future.

2. Where Are We Today?

As a precursor to routine remote high-speed access to large-scale mass storage systems, a recent set of experiments were conducted between Lawrence Berkeley National Laboratory (LBNL) in Berkeley, Calif., and the Stanford Linear Accelerator (SLAC) in Palo Alto, Calif. The National Transparent Optical Network testbed (NTON - see [1]) testbed provides eight 2.4 gigabit/sec data channels around the San Francisco Bay, of which four are usually used for OC-48 SONET. For this experiment, the network configuration involved four to six ATM switches and a Sun Enterprise-4000 SMP as a data receiver at

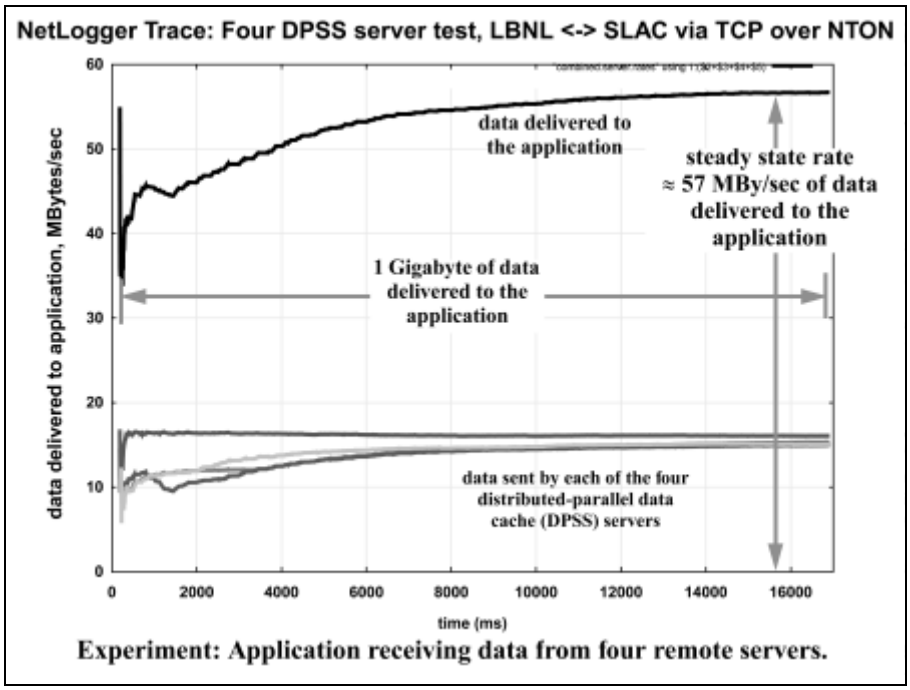

Figure 1 **Where we are now.**

[1]The work described in this paper is supported by the U. S. Dept. of Energy, Office of Energy Research, Office of Computational and Technology Research, Mathematical, Information, and Computational Sciences and ERLTT Divisions under contract DE-AC03-76SF00098 with the University of California, and by DARPA, Information Technology Office. This is report no. LBNL-41862. wejohnston@lbl.gov, www-itg.lbl.gov/~johnston

SLAC, all with OC-12 (622 Mbit/sec) network interfaces, and four smaller systems at LBNL configured as distributed caches and serving as data sources. The results of this experiment were that a sustained 57 megabytes/sec of data were delivered from datasets in the distributed cache to the remote application memory, ready for analysis algorithms to commence operation. The results of this experiment are presented in Figure 1.

This fairly impressive experiment is the result of a ten-year evolution of computing and networking technology, involving advances in platform and interface technologies, monitoring and management approaches, and parallel distributed software architectures and algorithms.

For the next few pages I will relate some of the history of the evolution, then return to the current circumstances that enable the results of Figure 1.

3. How Did We Get Here?

3.1 The Gore Demonstration: Selling the Potential

In the spring of 1989, then Senator Al Gore was holding hearings on his High Performance Computing and Communication legislation. At one of the early hearings, Craig Fields, then head of DARPA, was invited to provide testimony on the impact of high-speed networks. Through various circumstances, LBL was asked to provide a demonstration that would relate remote visualization and networking. A "live" network connection was ruled out (we were told that this exercise was the first computer demonstration in a Senate hearing room and they did not want to try for a network connection on top of everything else) so a realistic "simulation" was required. A collection of scientific visualization movies were put together, and, at the suggestion of Mark Pullen (DARPA), Steve Casner (then of ISI) and Van Jacobson (LBL) did various measurements on the new NSFNet T3 (45 megabits/sec) backbone. They measured packet delays on cross country connections, and those delays were then used to clock out the movie frames for display. A Sun Microsystems workstation with a graphics co-processor (required at that time to get even a local playback that was reasonably fast) was used to simulate transmission of the movie frames across networks of various speeds (19 kilobits/sec to 40 megabits/sec). The resulting video display of the movie gave the Senators an appreciation for implications of data network bandwidth.

3.2 Supercomputing 1991: Demonstrating the Potential

A demonstration at SC91 (in Albuquerque, NM) was arguably the first use of "high-speed" wide area networks to support a high-speed TCP/IP based distributed application.

The goal was to demonstrate real-time remote visualization of a large, complex scientific dataset. The approach was to use a Thinking Machines' CM-2 and Cray Y-MP at the NSF's Pittsburgh Supercomputer Center (PSC) to compute the visualization of a large medical dataset (a high-resolution MRI scan of a human brain). This type of data is essentially a 3D scalar field, and contours of this data represent surfaces of various types of brain tissue and structures. It is these surfaces that are identified and displayed. (See Figure 2.)

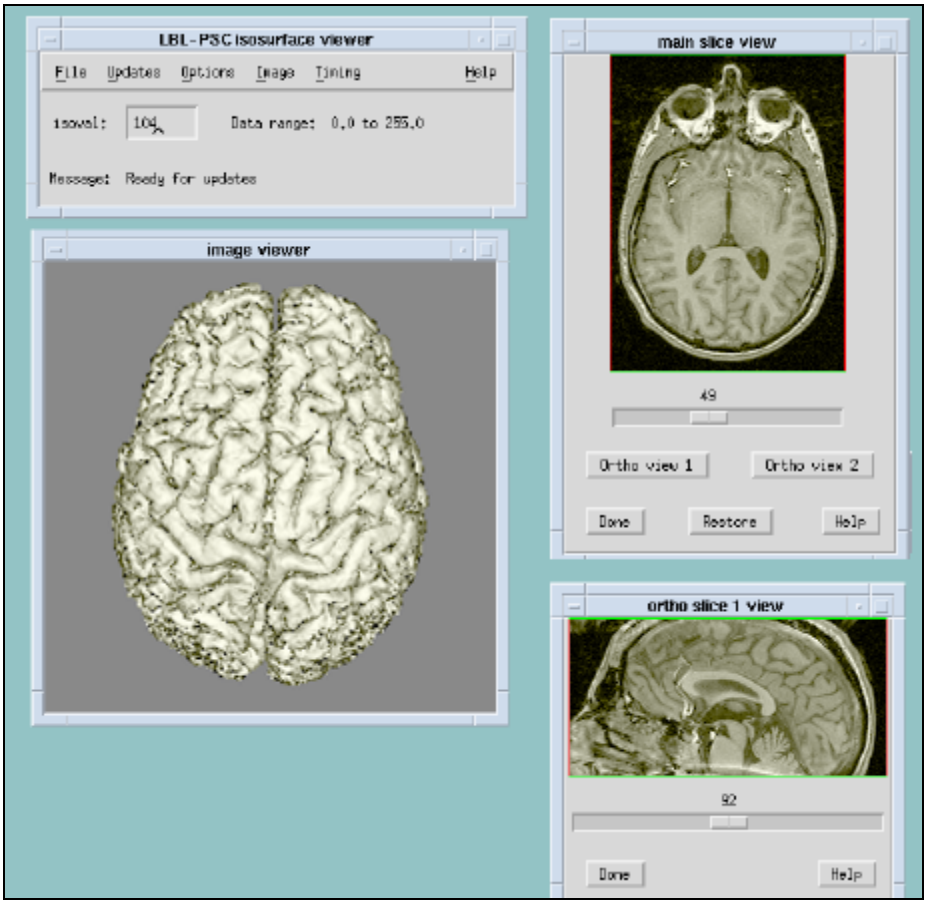

Figure 2 Remote, real-time visualization: PSC to SC91.

In order to achieve a real-time visualization, an early implementation of Lorensen and Cline's Dividing Cubes algorithm [2] was partitioned between the CM-2 and the a Cray Y-MP. A user at a workstation in Albuquerque selected contour values to isolate brain features of interest, and these values were sent to the CM-2. The CM-2 contoured the surface and computed the surface normal vectors. These quantities were then sent over a HIPPI network to the Cray where the selected surface was rendered (converted to a 2D image) using an in-memory framebuffer on the Cray. The resulting image (see Figure 2) was then sent through a TCP circuit from the Cray, through a local FDDI router, from there to an NSFNet T3 backbone switch in Pittsburgh. From Pittsburgh a T3 circuit went to another NSFNet switch on the show floor at SC91, through another FDDI interfaced router and then to the Sun workstation in the booth, where the image was displayed. The 15 (or so) Mbits/sec that we were able to achieve between the Cray and the Sun was sufficient to display about 10-12 frames/sec on the Sun. This rate turned out to be a good match for all of the components, because the CM-2 and Cray could produce the rendered images at 5-10 frames/sec. The details may be found in [3].

Typical of distributed applications, many components had to interoperate to produce a functioning system, an especially difficult task in a wide-area network. David Robertson and Brian Tierney (LBL), and Wendy Huntoon, Jamshid Mahdavi, and Matt Mathis (PSC) spent a lot of time getting the CM-2, the Cray, and the network to interoperate. Dave Borman (Cray) and Van Jacobson (LBL) were doing kernel hacking on the Cray and the Sun up to the hour that the SC91 exhibits opened in order to accomplish the first heterogeneous operation of the TCP large window option that made high-speed TCP possible between Albuquerque and Pittsburgh. (Less than 2 Mbit/sec were possible using the standard 64-kbyte TCP windows).

The enduring legacy of this work was the experience gained in building widely distributed systems and the TCP modifications that allowed high data rates in the wide area.

3.3 Blanca: Video as Data

The Blanca testbed - one of the national "gigabit" testbeds funded through CNRI - actually consisted of three parts. XUnet was Bell Labs' "Experimental University Network," a T3 ATM network that ran from Bell Labs in

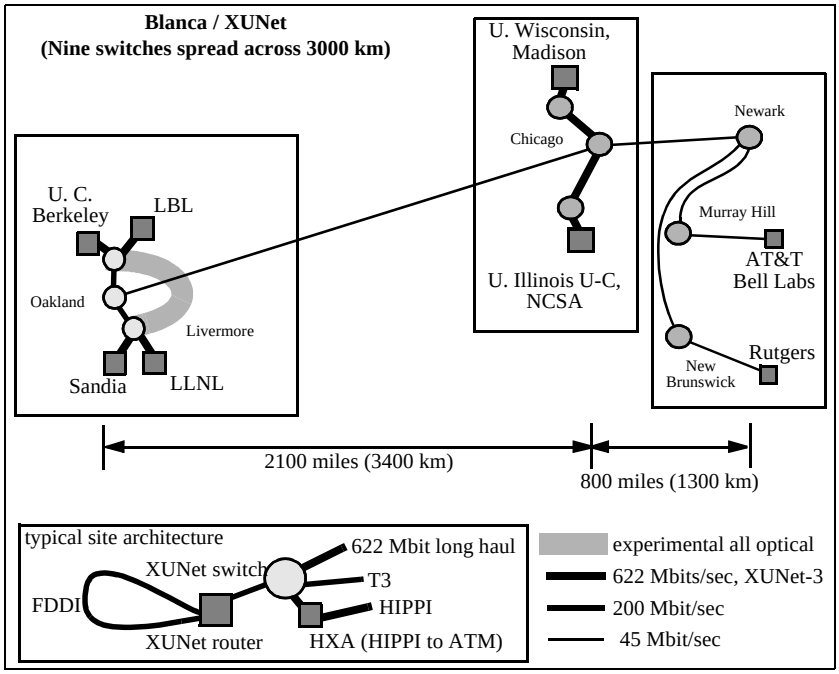

Figure 3 The Blanca/XUNet testbed, ca 1993.

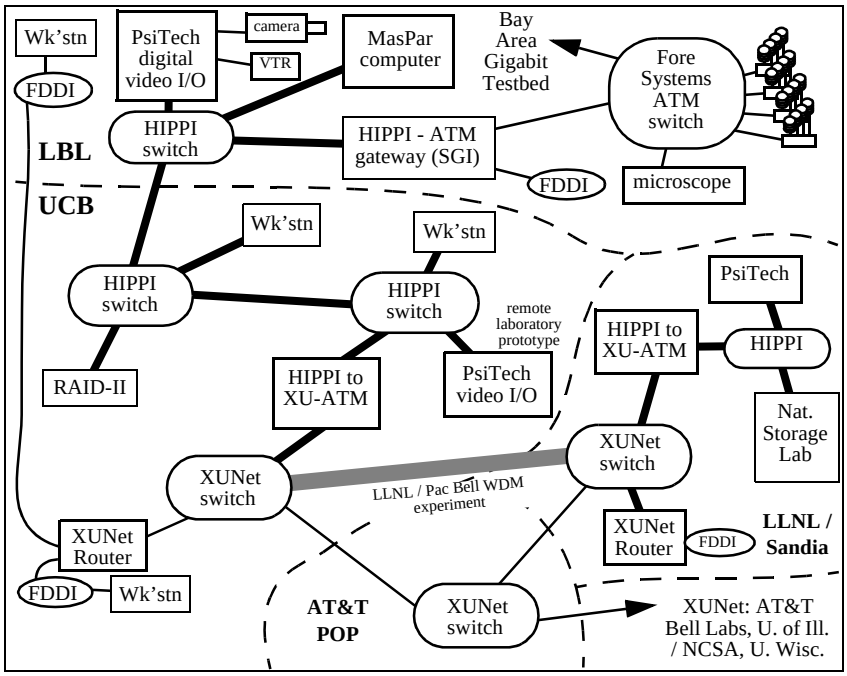

Figure 4 Remote dynamic experiment operation infrastructure in the Blanca-West/XUNet testbed.

New Jersey to Berkeley, Calif. Four universities - Rutgers, UW Madison, UI Urbana-Campaign, and UC Berkeley - together with Bell Labs, NCSA, and Sandia Livermore, were connected to the network. XUNet primarily supported ATM switch related research (queuing and management) at the Universities. Blanca was the "gigabit testbed" part of XUnet, and is the part to be discussed here.

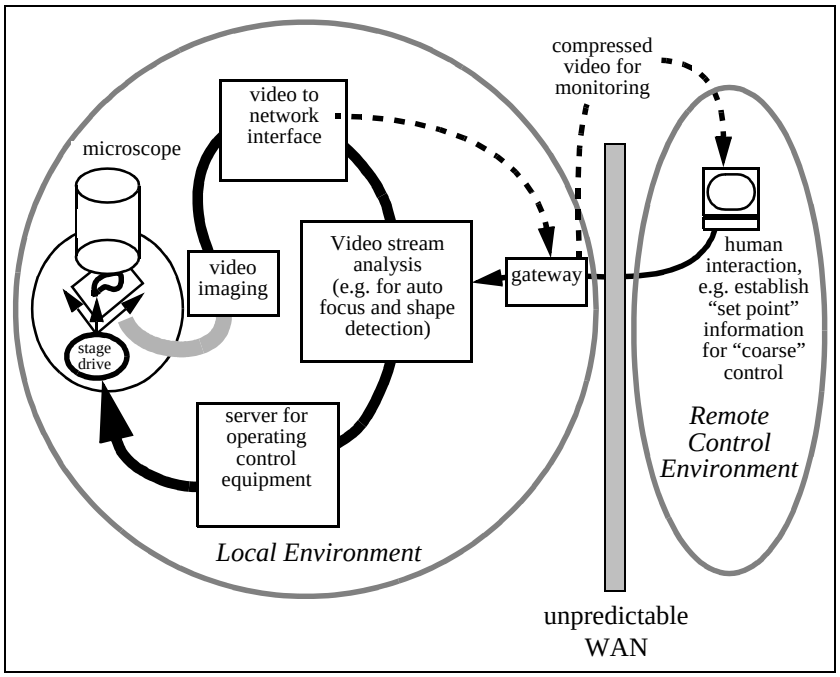

Figure 5 Remote, semi-autonomous, dynamic experiment operation architecture.

In Blanca, a HIPPI - ATM gateway connected HIPPI networks to the 622-Mbit/sec ATM interface on the XUNet switches. A 200 Mbit/sec XUNet ATM interface was built for a DEC workstation that acted as an IP router, and connected to FDDI and Ethernet networks. See Figure 3.

The data intensive aspect of the LBL-UCB part of Blanca ("Blanca West") was to develop the infrastructure and architecture needed to treat scientific (uncompressed) video as a routinely used data type, and then use this digital video as input for an instrument control system. This required video capture and conversion to digital form, high-speed transport through all of the components of the system, real-time analysis to extract information for understanding and control, visualization and display, and storage systems that could provide both long term and ready access.

Driving this work was the need to do real-time content analysis of video streams in order to enable the control of microscopic experiments through a process called visual servoing. Visual servoing uses information such as object shape change and velocity, obtained by analysis of the video images of the object, to provide the feedback needed for automated control of various microscopic regime instruments. The instruments, in turn, do things like cleaving DNA molecules [4] and controlling the shape of growing micro-crystals [5].

The Blanca/XUNet infrastructure that provided the capabilities mentioned above (Figure 4) required (at that time) a collection of HIPPI interfaced devices: a PsiTech HIPPI framebuffer did real-time video capture and display; a MasPar computer, with its revolutionary high-speed I/O architecture coupled with its massively parallel CPUs, enabled real-time analysis of the digital video; the experimental RAID-II system at UCB provided storage and retrieval of data at video rates; and the XUNet HIPPI-ATM gateway allowed real-time instrument control functions and compressed video monitoring at remote sites.

This all worked - with great difficulty, as is typical when interconnecting prototype systems - and microscopic visual servoing based on real-time content analysis of video was achieved for fleeting moments.

Though most of the specific technology approaches to the infrastructure have since been replaced by high-speed ATM networks and workstations with high bandwidth memory systems, the visual servoing architecture (Figure 5) has endured as a data intensive computing technology, and is used in several application regimes. The application of this architecture to micro-manipulation of DNA was ultimately awarded a patent. [6] The all-optical experiment indicated in Figure 3 and Figure 4 evolved into NTON.

3.4 BAGNet: Involvement of a Large Community and, Finally, a "Real" Application

BAGNet was an IP over OC-3 (155 Mbit/sec) ATM, metropolitan area testbed that operated in the San Francisco Bay Area (California) for two years starting in early 1994. The participants included government, academic, and industry computer science and telecommunications R&D groups from fifteen Bay Area organizations. The goal was to develop and deploy the infrastructure needed to support a diverse set of distributed applications in a large-scale, IP-over-ATM network environment. The participating organizations were Apple Computer, DEC – Palo Alto

Systems Research Center, Hewlett-Packard Laboratories, International Computer Science Institute, Lawrence Berkeley National Laboratory, Lawrence Livermore National Laboratory (LLNL), NASA Ames Research Center, Pacific Bell – Broadband Development Group, Sandia National Laboratories, Silicon Graphics, Inc., SRI International, Stanford University, Sun Microsystems, Inc., University of California, Berkeley, and Xerox Palo Alto Research Center (PARC).

The testbed consisted of a full-mesh, unicast ATM/PVC network supporting four end nodes at each of the fifteen sites, and a full-mesh ATM point-to-multipoint (multicast) link structure for each of the 15 sites. The unicast mesh provided an ATM "best-effort" quality of service over a 155-Mbit/sec SONET infrastructure between the (approximately) 60 connected systems. A single logical IP subnet overlaid this ATM network supporting a variety of distributed applications – see, for example, [7]. The ATM point-to-multipoint mesh was used to support IP multicast, and this capability supported high-quality multimedia teleseminars using the MBone tools: vic, vat, and wb. [8]

The PVC mesh consisted of about 1800 virtual circuits - a herculean management task accomplished by Berry Kercheval of Xerox PARC. The interior (central office) switches were primitive, and the whole network worked poorly until Lance Berc of DEC's Systems Research Center, Helen Chin of Sandia Livermore, and Dave Wiltzius of Lawrence Livermore National Lab identified a set of key central office ATM switch issues that Pacific Bell could address. (See [7].)

One of the overall testbed highlights was at ACM Multimedia94 when the ATM multicast mesh was used to include Steve McCanne in a conference panel session in San Francisco from his office at LBL. High-quality motion JPEG-encoded video used about 4 Mbit/sec of bandwidth to produce a spectacular, theater-sized and theater-quality image at the conference session as well as at other BAGNet sites (a temporary BAGNet connection was made to the conference hotel).

In addition to "community" projects in BAGNet, there were several specific projects involving subsets of the connected sites. In particular, LBNL, the Kaiser Permanente health care organization, and Philips Palo Alto Research Center collaborated to produce a prototype production, on-line, distributed, high data rate medical imaging system. (Philips and Kaiser were added to BAGNet for this project through the Pacific Bell CalREN program.)

The Kaiser project ([9] and [10]) focused on using high data rate, on-line instrument systems as remote data sources.

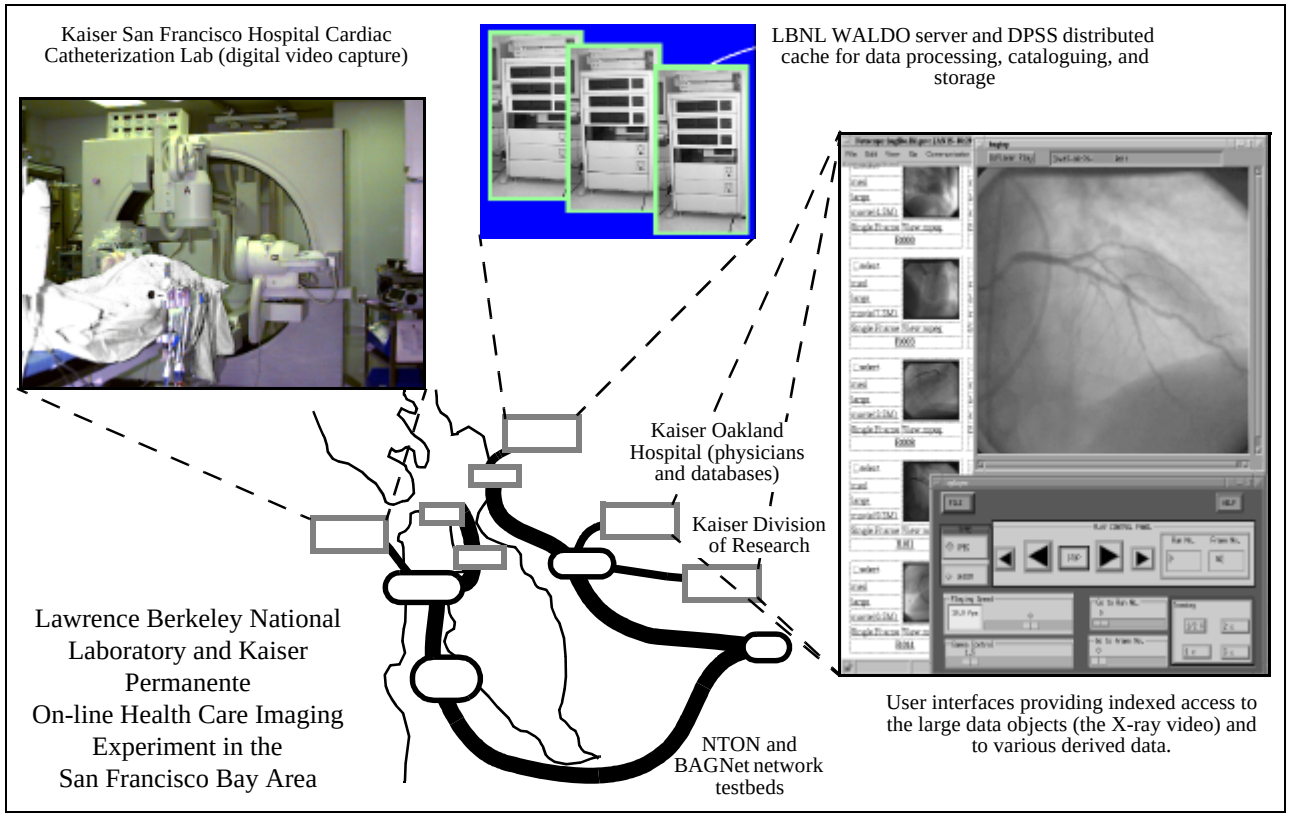

Figure 6 **A distributed, data-intensive health care imaging application.**

When data is generated in large volumes and with high throughput, and especially in a distributed environment where the people generating the data are geographically separated from the people cataloguing or using the data, there are several important considerations for managing instrument generated data:

- automatic generation of at least minimal metadata;
- automatic cataloguing of the data and the metadata as the data is received (or as close to real time as possible);
- transparent management of tertiary storage systems where the original data is archived;
- facilitation of cooperative research by providing specified users at local and remote sites immediate as well as long-term access to the data;
- mechanisms to incorporate the data into other databases or documents.

The WALDO (Wide-Area Large-Data-Object) system was developed to provide these capabilities, especially when the data is gathered in real time from a high data rate instrument. WALDO is a digital data archive that is optimized to handle real-time data. It federates textual and URL linked metadata to represent the characteristics of large data sets. Automatic cataloguing of incoming real-time data is accomplished by extracting associated metadata and converting it into text records; by generating auxiliary metadata and derived data; and by combining these into Web-based objects that include persistent references to the original data components (called large data objects, or LDOs). Tertiary storage management for the data components (i.e., the original datasets) is accomplished by using the remote program execution capability of Web servers to manage the data on mass storage systems. For subsequent use, the data components may be staged to a local disk and then returned as usual via the Web browser, or, as is the case for several of our applications, moved to a high-speed cache for access by specialized applications (e.g., the high-speed video player illustrated in the right-hand part of the right-hand panel in Figure 6). The location of the data components on tertiary storage, how to access them, and other descriptive material are all part of the LDO definition. The creation of object definitions, the inclusion of "standardized" derived-data-objects as part of the metadata, and the use of typed links in the object definition, are intended to provide a general framework for dealing with many different types of data, including, for example, abstract instrument data and multi-component multimedia programs. See [9].

WALDO was used in the Kaiser project to build a medical application that automatically manages the collection, storage, cataloguing, and playback of video-angiography data[1] collected at a hospital remote from the referring physician.

Using a shared, metropolitan area ATM network and a high-speed distributed data handling system, video sequences are collected from the video-angiography imaging system, then processed, catalogued, stored, and made available to remote users. This permits the data to be made available in near-real time to remote clinics (see Figure 6). The LDO becomes available as soon as the catalogue entry is generated — derived data (e.g. MPEG versions of the instrument digital video) is added as the processing required to produce it completes. Whether the storage systems are local or distributed around the network is entirely a function of optimizing logistics.

In the Kaiser project, cardio-angiography data was collected directly from a Philips scanner by a computer system in the San Francisco Kaiser hospital Cardiac Catheterization Laboratory. This system is, in turn, attached to an ATM network provided by the NTON and BAGNet testbeds. When the data collection for a patient is complete (about once every 20–40 minutes), 500–1000 megabytes of digital video data is sent across the ATM network to LBNL (in Berkeley) and stored first on the DPSS distributed cache (described below), and then the WALDO object definitions are generated and made available to physicians in other Kaiser hospitals via BAGNet. Auxiliary processing and archiving to one or more mass storage systems proceeds independently. This process goes on 8–10 hours a day.

WALDO provides the Web-based user interface to the data and to appropriate viewing applications. Hospital department-level Web-based patient databases can then refer directly to the data in WALDO without duplicating that data, or being concerned about tertiary storage management (which is handled by WALDO). This project is still active at Kaiser, where it is being investigated for use in a production environment.

The legacy of this project for data intensive environments is the general model for data intensive computing that is described below.

3.5 MAGIC: the First "Real" Data Intensive Environment

The MAGIC Gigabit testbed[2] is a DARPA-funded collaboration working on distributed applications in large-scale, high-speed, ATM networks. It is a heterogeneous collection of ATM switches and computing platforms, several different implementations of IP over ATM, a

1. Cardio-angiography imaging involves a two plane, X-ray video imaging system that produces from several to tens of minutes of digital video sequences for each patient study for each patient session. The digital video is organized as tens of data-objects, each of which are of the order of 100 megabytes.

collection of "middleware" (distributed services), etc., all of which must cooperate in order to make a complex application operate at high speed.

Another key aspect of a data intensive computing environment has turned out to be a high-speed, distributed cache. LBNL designed and implemented the Distributed-Parallel Storage System (DPSS) as part of the MAGIC project, and as part of the U.S. Department of Energy's high-speed distributed computing program. This technology has been quite successful in providing an economical, high-performance, widely distributed, and highly scalable architecture for caching large amounts of data that can potentially be used by many different users. In the MAGIC testbed a multi-server DPSS is typically distributed across several sites separated by more than 2600 km of high-speed, IP-over-ATM network, and is used to store very high resolution images of several geographic areas. The first application use of the DPSS was *TerraVision*, a terrain visualization application that uses the DPSS to let a user explore / navigate a "real" landscape represented in 3D by using ortho-corrected, one meter per pixel images and digital elevation models (see [13]). TerraVision requests from the DPSS, in real time, the sub-images ("tiles") needed to provide a view of a landscape for an autonomously "moving" user. Typical use requires aggregated data rates as high as 100 to 200 Mbits/sec. The DPSS is easily able to supply these data rates from several disk servers distributed across the network.

The combination of the distributed nature of the DPSS, together with the high data rates required by TerraVision and various load simulators, makes the DPSS a good system with which to test high-speed data intensive environments.

A central issue for using high-speed networks and widely distributed systems as the foundation of a large data-object management strategy is the performance of the system components, the transport and OS software, and the underlying network. Problems in any of these regimes will hinder a data intensive computing strategy, but such problems can usually be corrected if they can be isolated and characterized.

There are virtually no behavioral aspects of high-speed, wide area IP-over-ATM networks that can be taken for granted, even in end-to-end ATM networks. By "network" we mean the end-to-end data path from the transport API through the host network protocol (TCP/IP) software, the host network adaptors and their device drivers, the many different kinds of ATM switches and physical links, up through the corresponding software stack on the receiver. Further, the behavior of different elements at similar places in the network architecture can be quite different because they are implemented in different ways. The combination of these aspects can lead to complex and unpredictable network behavior.

A significant part of our work with high-speed distributed systems in MAGIC has been developing an agent-based monitoring methodology and associated tools to locate and characterize bottlenecks. We have used these performance and operation monitoring tools in the storage system and several applications to characterize the distributed operation of the system at many levels. As requests and data enter and leave all parts of the user-level system, synchronized timestamps are logged using a common logging format. At the same time, various operating system and network parameters may be logged in the same format. This is accomplished by the Netlogger monitoring system [14], which has been used to analyze several network-generated problems that showed up in the distributed applications [15].

The DPSS serves several roles in high-performance, data-

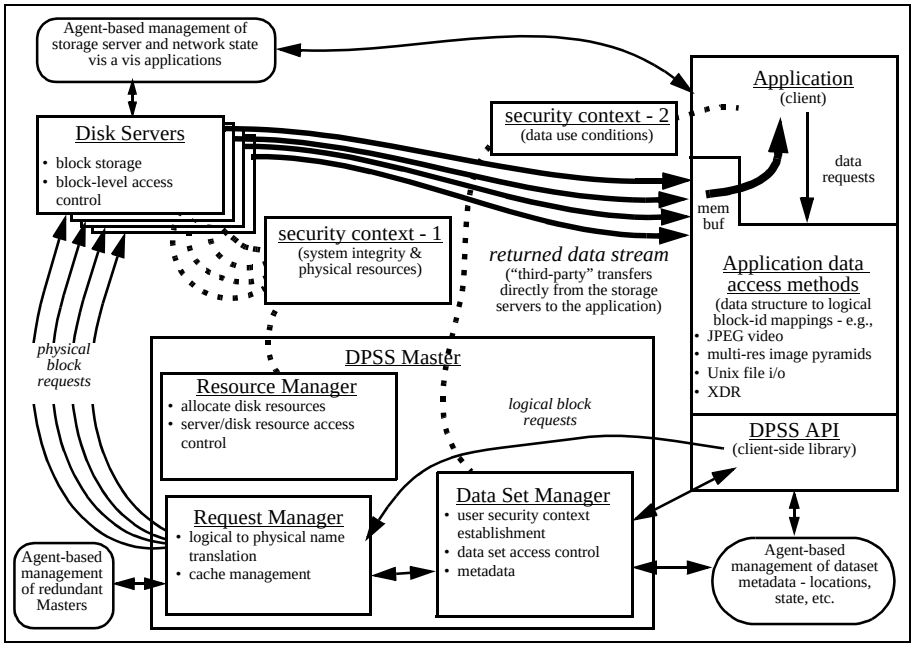

Figure 7 Distributed-Parallel Storage System architecture.

2. MAGIC was established in June 1992 ment's Defence Advanced Research Proje The testbed is a collaboration between LE computer Center, SRI, Univ. of Kansas, I EROS Data Center, CNRI, Sprint, and S [11] and [12].

intensive computing environments. This application-oriented cache provides a standard high data rate interface for high-speed access by data sources, processing resources, mass storage systems (MSS), and user interface elements. It provides the functionality of a single very large, random access, block-oriented I/O device (i.e., a "virtual disk") with very high capacity (we anticipate a terabyte sized system for high-energy physics data) and serves to isolate the application from tertiary storage systems and instrument data sources. Many large data sets may be logically present in the cache by virtue of the block index maps being loaded even if the data is not yet available. In this way processing can begin as soon as the first data blocks are generated by an instrument or migrated from tertiary storage.

The naming issues (e.g., resolving independent name space conflicts) are handled elsewhere. For example, in the on-line health care imaging system mentioned above, the name space issue is addressed by having all of the data represented by Web-based objects which are managed by WALDO. At the minimum, WALDO provides globally unique naming and serves as a mechanism for collecting different sources of information about the data. It also manages object use-conditions through a PKI access control system - see [16].

The DPSS provides several important and unique capabilities for the data intensive computing environment. It provides application-specific interfaces to an extremely large space of logical blocks; it may be dynamically configured by aggregating workstations and disks from all over the network (this is routinely done in the MAGIC testbed, and it will in the future be mediated by an agent-based management system); it offers the ability to build large, high-performance storage systems from inexpensive commodity components; and it offers the ability to increase performance by increasing the number of parallel disk servers. Various cache management policies operate on a per-data set basis to provide block aging and replacement.

The high performance of the DPSS - about 10 megabytes/sec of data delivered to the user application per disk server - is obtained through parallel operation of independent, network-based components. Flexible resource management - dynamically adding and deleting storage elements, partitioning the available storage, etc. - is provided by design, as are high availability and strongly bound security contexts. The scalable nature of the system is provided by many of the same design features that provide the flexible resource management (that in turn provides the capability to aggregate dispersed and independently owned storage resources into a single cache).

While the basic data access interface provides for requesting lists of named logical blocks, many applications use file-like or other I/O semantics, and these are provided in the DPSS client-side interface library.

The (ongoing) legacies of MAGIC and DPSS for data intensive computing are their establishment of the importance of a high-speed distributed cache, the agent based monitoring and management architecture, the highly distributed security architecture, and the development of "global" data management strategies.

3.6 Where We Are Today, Revisited: An Overall Model for Data-Intensive Computing

The concept of a high-speed distributed cache as a common element for all of the sources and sinks of data involved in high-performance data systems has proven very successful in several application areas, including the automated processing and cataloguing of real-time instrument data and the staging of data from an MSS for high data-rate applications.

For the various data sources and sinks, the cache, which

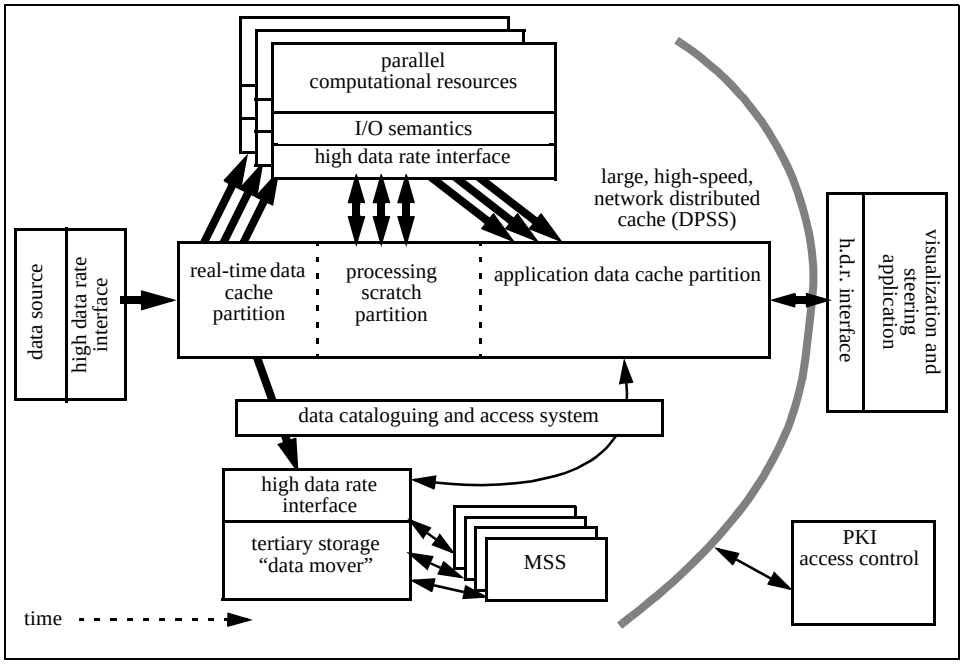

Figure 8 **Data intensive computing model.**

is itself a complex and widely distributed system, provides:

- a standardized approach for high data-rate interfaces;
- an "impedance" matching function (e.g., between the coarse-grained nature of parallel tape drives in the tertiary storage system and the fine-grained access of hundreds of applications);
- flexible management of on-line storage resources to support initial caching of data, processing, and interfacing to tertiary storage;
- a unit of high-speed, on-line storage that is large compared to the available disks of the computing environments, and very large (e.g., hundreds of gigabytes) compared to any single disk.

The model for data intensive computing, then, includes the following:

- each application uses a standard high data-rate interface to a large, high-speed, application-oriented cache that provides semi-persistent, named datasets / objects
- data sources deposit data in a distributed cache, and consumers take data from the cache, usually writing processed data back to the cache when the consumers are intermediate processing operations
- metadata is typically recorded in a cataloguing system as data enters the cache, or after intermediate processing
- a tertiary storage system manager typically migrates data to and from the cache. The cache can thus serve as a moving window on the object/dataset, since, depending on the size of the cache relative to the objects of interest, only part of the object data may be loaded in the cache - though the full objection definition is present: that is, the cache is a moving window for the off-line object/data set
- the native cache access interface is at the logical block level, but client-side libraries implement various access I/O semantics - e.g., Unix I/O (upon request available data is returned; requests for data in the dataset, but not yet migrated to cache, cause the application-level read to block or be signaled)

This model is illustrated in Figure 8.

The LBNL-SLAC-NTON High Data-Rate Experiments

Brian Tierney, Jason Lee, Craig Tull, and Bill Johnston (LBNL); Les Cottrell and Dave Millsom (SLAC); Bill Lennon and Lee Thombley (LLNL/NTON), with support from Hal Edwards (Nortel), conducted a series of experiments in high-speed, wide area distributed data processing that represent an example of our data intensive computing model in operation.

The prototype application was the STAR analysis system that analyzes data from high energy physics experiments. (See [17].)

A four-server DPSS located at LBNL was used as a prototype front end for a high-speed mass storage system. A 4-CPU Sun E-4000 located at SLAC was a prototype for a physics data analysis computing cluster. The NTON network testbed that connects LBNL and SLAC provided a five-switch, 100-km, OC-12 path (and could be configured as a 2000 km, OC-12, path). All experiments were application-to-application, using TCP transport.

Multiple instances of the STAR analysis code read data from the DPSS at LBNL and moved that data into the memory of the STAF application where it was available to the analysis algorithms. This experiment resulted in a sustained 57 MBytes/sec of DPSS cache to application memory data transfer, as indicated in Figure 1. The goal of the experiment was to demonstrate that high-speed mass

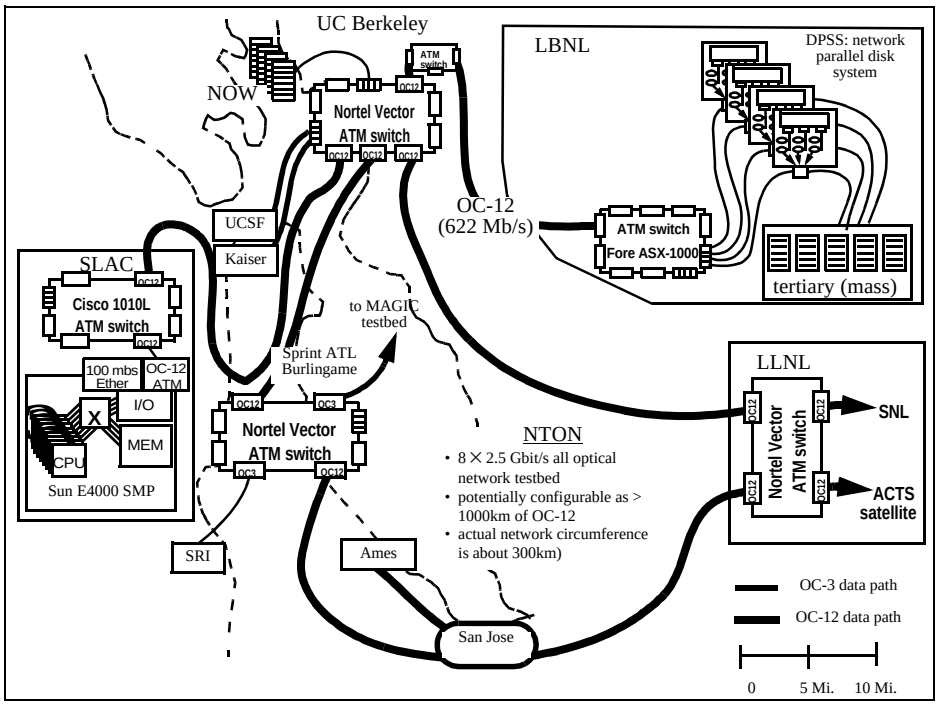

Figure 9 The LBNL - NTON - SLAC high-speed, data intensive distributed computing experiment.

storage systems could use distributed caches to make data available to the systems running the analysis codes. The experiment was successful, and the next steps will involve completing the mechanisms for optimizing the MSS staging patterns and completing the DPSS interface to the bit file movers that interface to the MSS tape drives.

Additionally, in the context of the network R&D on NTON, these experiments raised a number of other questions to be addressed:

- What are the issues with the routine use of remote network disks to support high data-rate environments?
  - Are there any additional issues if the network cache is the front end for a high-performance mass storage system (HPSS)?
  - How well does TCP congestion control work for high-speed data streams in high-speed networks?
  - How do we do high-speed rate shaping for QoS?
- How do anomalies in a separate control channel connection impact performance (as might be encountered if control information were going over a ground line while data was going over a high bandwidth-delay product link like the ACTS satellite)?
- How does striping across multiple, independent physical paths impact performance in a parallel-distributed system?

In addition to the advances in architecture and software, the success of this experiment is due to the "third generation" platform architectures like the Sun UltraSPARC (e.g. the Enterprise 4000) that provide gigabyte/sec memory bandwidth and OC-12 ATM interfaces that actually pass data through at the full rate.

4. NGI: What Comes Next?

The U.S. Dept. of Energy operates some of the world's largest, on-line, shared scientific instrument systems. These include high voltage electron microscopes, synchrotron light sources, high energy particle accelerators, gigahertz NMR systems, special telescopes, quartz piston diesel engines, and so on. These instrument systems - many of which are national user facilities - have a wide variety of computing and storage requirements associated with them, but they share certain characteristics that motivate much of our work in data intensive computing. These instruments are used by people in research labs and universities all over the world. It has already been shown in various collaboratory[3] projects that remote users will gain significant efficiencies through transparent remote access involving widely distributed data intensive computing in almost every case. See, for example, [4], [5], [18], and [19].

However, the major scientific gains are likely to come when we can couple these instruments to large-scale computing and storage systems. In most cases, observation and experimentation advances our knowledge when we are able to match experiment and theory. Today this is dramatically slowed by the long time required to collect data and do off-line processing in order to compare experiments with computational models that represent the current theory of the underlying physical processes. Both the experiment and the model are adjusted and the process repeated. The process of analyzing experiments and comparing them with theory is almost always hampered by the lack of sufficient storage and computation.

Development of flexible, transparent, and dynamic data intensive computing environments will provide a big step forward by simultaneously coupling experiments and instruments to large scale computing, and by providing greater capacity for simulation. Ultimately, the flexibility and transparency of data intensive environments will allow scientific experiments to be directly coupled to the computation simulations of the subject phenomenon through the use of high-speed networks, distributed storage systems and computation that can be scheduled so that all of the required resources are available to match small windows of instrument operation time, etc. When this happens, the weeks of off-line computational analysis of experiment data followed by the (largely manual) feedback of experimental results into the models, and vice versa, should be greatly shortened, permitting more and different kinds of experiments, more accurate and detailed insights, etc.

This vision of cooperative operation of scientific experiments and computational simulation of the theoretical models is one of the ultimate goals for our work in data intensive computing.

The China Clipper project[4] - a collaboration between LBNL, SLAC, and Argonne National Lab that builds on the infrastructure of the Energy Sciences network [20] - is the continuation of our years of work in this area, and has as its

3. "The fusion of computers and electronic communications has the potential to dramatically enhance the output and productivity of U. S. researchers. A major step toward realizing that potential can come from combining the interests of the scientific community at large with those of the computer science and engineering community to create integrated, tool-oriented computing and communication systems to support scientific collaboration. Such systems can be called 'collaboratories.'" From "National Collaboratories - Applying Information Technology for Scientific Research," Committee on a National Collaboratory, National Research Council. National Academy Press, Washington, D. C., 1993.

4. Like Pan American Airway's historic China Clipper that made the first trans-Pacific airmail flights from San Francisco to Honolulu and Manila, and its companion "flying boats" at the beginning of large-scale airline service, the Clipper Project anticipates the future in both flexibility and performance.

high level goals designing and implementing a collection of independent but architecturally consistent service component. This is intended to enhance the ability of a variety of applications and systems to construct and use distributed, high-performance infrastructure. Such middleware will support high-speed access to and integrated views of multiple data archives; resource discovery and automated brokering; comprehensive real-time monitoring and performance trend analysis of the networked subsystems, including the storage, computing, and middleware components; and flexible and distributed management of access control and policy enforcement for multi-administrative domain resources.

Adaptability is an important aspect of distributed environments. Critical subsystems and components (e.g., network caches) must be capable of dynamic reconfiguration in a manner that is transparent to the application. Applications will also have to make use of performance trend information from the distributed components, and dynamically optimize their behavior.

The challenge addressed by the Clipper project is how to accelerate routine use of applications that:

- require substantial computing resources
- generate and/or consume high rate and high volume data flows
- involve human interaction
- require aggregating many dispersed resources to establish an operating environment:
  - multiple data archives
  - distributed computing capacity
  - distributed cache capacity
  - "guaranteed" network capacity
- operate in widely dispersed environments.

Our general approach to addressing this challenge involves a combination of architecture, network functionality, middleware, and their integration into prototype applications. This project will develop network and middleware capabilities and prototype applications that provide and demonstrate environments for routine creation of robust, high data rate, secure distributed systems. Project objectives include:

- high-speed network connectivity that offers schedulable quality of service by effectively providing some level of bandwidth reservation among the elements of distributed systems
- data management architectures that provide federated views of and high-performance "external" access to multiple archival mass storage systems
- distributed, high-speed caches that provide applications with very high-performance access to data that is collected from on-line scientific instruments, staged from mass storage systems (MSS), or is in various stages of analysis, regardless of the physical location of the data or computing elements
- resource monitoring to support problem diagnosis and infrastructure performance experiments, and to provide performance trend indicators to adaptive applications
- active management of distributed elements to provide fault-tolerant operation of distributed system components and software
- a security infrastructure that enforces the use and scheduling agreements among the services that are required by an application, and that also provides strong access control

Clipper is envisioned not so much as a "system" but rather as a coordinated collection of services that may be flexibly employed by a variety of applications (or other middleware) to build on-demand, large-scale, high-performance, wide area, problem-solving environments.

The Clipper project and its integration with, and support of, systems like Globus ([21], [22]), WALDO, DPSS, Netlogger, and SRB [23], will enable the next generation of configurable, distributed, high-performance, data-intensive systems; computational steering; and integrated instrument and computational simulation.

5. Acknowledgments

Many people have contributed to this work in many different ways. Some of those people are identified in the text, and some in the citations. However, some other comments are necessary. Stewart C. Loken, head of the Information and Computing Sciences Division at LBNL has been a longtime supporter of this work, and as a high energy physicist he has a special appreciation for its potential. Originally John Cavallini, and then Bob Aiken and Dan Hitchcock of the DOE, Energy Research, Computer Science program office have been sufficiently convinced of the worth of the approach to fund it for the past decade. I have always considered the MAGIC network testbed project to be one of the most successful of its type in contributing to the field of data intensive computing, and this is due not just to DARPA and DOE funding, but the many technically excellent people who worked on the project and toured the brew pubs of the midwest. Special thanks goes to Ira Richer, the long time project leader for MAGIC, with his ability to "herd cats" so effectively. Brian Tierney, Jason Lee, Gary Hoo, Jin Guojun, and Mary Thompson of LBNL have provided the implementation expertise to "make it all happen." AT&T, Bob Kahn (CNRI), and Pacific Bell contributed to many of the network testbed environments where the work was proven (or not). Special thanks goes to Sprint (John Strand and Mike Sobek, in particular) for long term support of, and direct participation in, MAGIC and NTON -in my opinion the two most important and effective

high-speed, wide area network testbeds. The DOE ESNet provides the prototype production environment where R&D ideas are introduced into the production scientific environment. Finally, without Van Jacobson, Sally Floyd, and their colleagues, we would not have the basic network mechanisms on which we have built much of our success.

6. References

[1] NTON, "National Transparent Optical Network Consortium." See http://www.ntonc.org.

[2] Cline, H.E., Lorensen W.E., Ludke, S., Crawford, C.R., and Teeter, B.C. Two algorithms for the three-dimensional reconstruction of tomograms, Medical Physics 15 3 (May/June 1988), 320-327.

[3] Johnston, W., V. Jacobson, S. C. Loken, D. W. Robertson, and B. L. Tierney, "High-Performance Computing, High-Speed Networks, and Configurable Computing Environments: Progress Toward Fully Distributed Computing," in *High-performance Computing in Biomedical Research*, T. Pilkington, et al, eds. CRC Press, 1993.

[4] Parvin, B., D. E. Callahan, W. Johnston, and M. Maestre, "Visual Servoing for Micro Manipulation," International Conference on Pattern Recognition, August. 1996. (Available at http://www-itg.lbl.gov/ITG.hm.pg.docs/VISION/vision.html)

[5] Parvin, B., J. Taylor, D. E. Callahan, W. Johnston, and U. Dahmen, "Visual Servoing for Online Facilities," IEEE Computer July 1997. (Available at http://www-itg.lbl.gov/ITG.hm.pg.docs/VISION/vision.html)

[6] Parvin, Bahram A., Marcos F. Maestre, Richard H. Fish, William E. Johnston, "A Method and Apparatus for Accurately Manipulating and Object During Microelectrophoersis." US Patent No. 423,969.

[7] Wiltzius, D., L. Berc, and S. Devadhar, "BAGNet: Experiences with an ATM metropolitan-area network," ConneXions, Volume 10, No. 3, March 1996. Also see http://www.llnl.gov/bagnet/connexions.html .

[8] LBNL Network Research Group, "vic, vat, wb, and sd," the MBone (IP multicast) teleconferencing tools, described at http://ee.lbl.gov.

[9] Johnston, W., G. Jin, C. Larsen, J. Lee, G. Hoo, M. Thompson, B. Tierney, J. Terdiman, "Real-Time Generation and Cataloguing of Large Data-Objects in Widely Distributed Environments," International Journal of Digital Libraries - Special Issue on "Digital Libraries in Medicine". November, 1997. (Available at http://www-itg.lbl.gov/WALDO)

[10] Thompson, M., W. Johnston, J. Guojun, J. Lee, B. Tierney, and J. F. Terdiman, "Distributed health care imaging information systems," PACS Design and Evaluation: Engineering and Clinical Issues, SPIE Medical Imaging 1997. (Available at http://www-itg.lbl.gov/Kaiser.IMG)

[11] Fuller, B., I. Richer "The MAGIC Project: From Vision to Reality," IEEE Network, May, 1996, Vol. 10, no. 3.

[12] MAGIC, "The MAGIC Gigabit Network", See: http://www.magic.net

[13] Lau, S, and Y. Leclerc, "TerraVision: a Terrain Visualization System,", Technical Note 540, SRI International, Menlo Park, CA, Mar. 1994. (http://www.ai.sri.com/~magic/terravision.html)

[14] Tierney, B., W. Johnston, B. Crowley, G. Hoo, C. Brooks, D. Gunter, "The NetLogger Methodology for High Performance Distributed Systems Performance Analysis," Seventh IEEE International Symposium on High Performance Distributed Computing, Chicago, Ill., July 28-31, 1998. Available at http://www-itg.lbl.gov/DPSS/papers.html .

[15] Tierney, B., W. Johnston, J. Lee, and G. Hoo, "Performance Analysis in High-Speed Wide Area ATM Networks: Top-to-bottom end-to-end Monitoring," IEEE Networking, May 1996. (Available at http://www-itg.lbl.gov/DPSS/papers .)

[16] Johnston, W., S. Mudumbai, M. Thompson, "Authorization and Attribute Certificates for Widely Distributed Access Control," in Proceedings of the Third International Workshop on Enterprise Security, Stanford University, June 17-19, 1998 as a part of the 7th Intl. Workshop on Enabling Technologies - Infrastructure for Collaborative Technologies (WETICE'98). (Available at http://www-itg.lbl.gov/Security/Akenti .)

[17] Greiman, W., W. E. Johnston, C. McParland, D. Olson, B. Tierney, C. Tull, "High-Speed Distributed Data Handling for HENP," Computing in High Energy Physics, April, 1997. Berlin, Germany. (Available at http://www-itg.lbl.gov/STAR)

[18] Agarwal, D., at al, "The Spectro-Microscopy Collaboratory at the Advanced Light Source." http://www-itg.lbl.gov/BL7Collab

[19] Johnston, W., et al, "The Distributed, Collaboratory Experiment Environments (DCEE) Program." http://www-itg.lbl.gov/DCEE

[20] ESNet, "The Energy Sciences Network," www.es.net. ("ESnet provides global networking for the DOE research and development mission. We are a leader in internet design and innovation providing a major piece of the U.S. Internet backbone.")

[21] Foster, I., C. Kesselman, eds., "The Grid: Blueprint for a New Computing Infrastructure," Morgan Kaufmann, publisher. August, 1998.

[22] Globus, "The Globus Project," http://www.globus.org

[23] Moore, R., et al, "Massive Data Analysis Systems," San Diego Supercomputer Center. See http://www.sdsc.edu/MDAS

